

# Design of an Optimal Route Structure Using Heuristics-Based Stochastic Schedulers

Seongim Choi

*University Affiliated Research Center, NASA Ames Research Center, Moffett Field, CA 94035.*

John E. Robinson III and Daniel G. Mulfinger

*NASA Ames Research Center, Moffett Field, CA 94035.*

Brian J. Capozzi

*Mosaic ATM, Inc., Leesburg, VA 20175, USA*

## Abstract

The purpose of this study is to investigate the effects of efficient route structure in the extended terminal airspace area on arrival scheduling performance. This paper will provide reasonable guidelines for optimal route topology in the extended terminal area by considering the uncertainties present in real operations. In a previous study, a Mixed Integer Linear Programming (MILP)-based scheduling algorithm proved to generate more optimal scheduling results than a traditional First-come-First-Served (FCFS) scheduler. However, an expensive computational cost associated with extensive search process limited its usage to a small number of flights in a dense terminal environment. Heuristics based on FCFS scheduling were introduced to alleviate this computational limitation. However, that heuristic was not sufficient to accommodate the amount of traffic associated with dense terminal operations. In this study, we introduce a Genetic Algorithm (GA) as an alternative heuristic for queuing aircraft and route assignment to reduce the computational cost dramatically. To take into account realistic operations, a dynamic planner framework is constructed that integrates the GA heuristics-based scheduler with a stochastic trajectory simulator. Uncertainty quantification and propagation along the routes are implemented in the trajectory model. The trajectory model is simulated based on the Scheduled Times of Arrival (STAs) provided by the scheduler. As

a practical application of the proposed scheduler to the dense terminal environment, a design of an optimal route structure is carried out for the terminal airspace represented in cartesian coordinates. The effects of airspace topologies on the scheduling performance are investigated and numerous route structures with different merge topologies are constructed. An optimal merge topology is identified by comparing their scheduling performances and the resulting optimal route structure is validated by the dynamic planner framework. Finally, the sensitivities of the scheduling performance with respect to the uncertainty quantification and propagation modeling are discussed.

## Introduction

A concept of advanced terminal airspace area operations is part of the Next Generation Air Transportation System (NextGen) efforts to accommodate expected increase in the demand and provide a higher level of throughput at the airports and within en route airspace [1, 2, 3]. Scheduling optimization problems in dense terminal airspace area operations are drawing interest [4], since the increase of airspace capacity is becoming more important to accommodate large air traffic flows in that environment. The use of scheduling algorithms more efficient than the traditional FCFS approach is a way to increase a way to increase throughput and efficiencies in congested terminal airspace compared to the FCFS scheduler. On the

other hand, it is known that the route topologies in the extended terminal airspace play an important role in the scheduling performance. Efficient scheduling and route assignment directly affect important performance metrics such as runway delays, throughput, fuel efficiency, and robustness to uncertainties in operations. However, despite the realization of the importance of the route topology, few extensive studies of the problem have been reported.

The purpose of this paper is: 1) to investigate optimal and efficient scheduling algorithms for a dense terminal airspace operation that yield better performance compared to the traditional FCFS approach, and 2) to design an optimal route structure for the extended terminal area.

A scheduling algorithm based on MILP has been successfully used in airport surface management due to its ability to optimize both the arrival/departure sequence and their scheduling [5]. However, due to the expensive computational cost attributed to the branch and bound search algorithm, only a limited number of flights, typically less than 40, were allowed in the scheduling. The application of MILP-based scheduling algorithms to the dense terminal airspace operation has been less effective. Consideration of hundreds of flights in the span of a couple of hours typical of the dense terminal airspace operation dramatically increases the size of the MILP formulation and the corresponding computation burden easily exceeds the available computing resources. In a previous study, a heuristics-based MILP formulation was introduced in order to reduce computational burden and applied it to metroplex operations during their busiest operations [6]. However, the cost for the branch and bound search is still expensive though not as prohibitive as in the original MILP formulation.

The combination of GA-based heuristics and a Linear Programming (LP) method was proposed by Capozzi et al. [8] This is the method that is adopted in the current work. It explicitly separates the search for the optimum binary variables of route assignment and aircraft sequencing from the solution procedure for the continuous scheduling variables. GA-based MILP optimization scheme is able to find the optimal solution in significantly less computational time on the example problem considered.

A central focus of the current work is the design of an optimal route topology in the extended terminal area using the GA-based MILP scheduler. This work will also use with flight trajectory uncertainty which is different from the previous work that held an unrealistic premise that detailed flight intent information including the transit times is known a priori. This would assume that one snapshot of the planning is sufficient to predict the scheduling performance. However, this is not true in the real-time simulation where uncertainties are present in flight trajectories and the schedules of the following flights are dynamically updated later in time based on the schedules of the previous flights.

In order to overcome the above issues, we developed a dynamic planner framework that periodically updates the flight schedules to handle uncertainties. The dynamic planner consists of separate modules: a planner and a simulator. Uncertainty is implemented in the trajectory model of the simulator to account for aircraft arrival time errors. The planner adds an extra separation buffer at the scheduling points to cope with these inter-arrival errors.

As a practical application of the heuristics-based, stochastic schedulers to dense terminal airspace simulation, a design of an optimal route structure in the extended terminal airspace area is carried out. Key design parameters are the number of merge points and their locations. First, we construct five distinct route structures with various merge topologies. The scheduling performance is evaluated for each topology using the stochastic FCFS heuristics-based scheduler and a dynamic planner framework is applied to each topology in order to validate the predicted scheduling performance. It is demonstrated that the largest separation amount required at all scheduling points is a dominant factor in scheduling performance.

Finally, more generalized airspace topologies are considered. To investigate the sensitivities of the merge topology to the uncertainty modeling, three types of uncertainty distributions are considered: a constant, linear and quartic increment of the uncertainty per unit route length. The comparison of the corresponding scheduling results show that: 1) there exist optimal merge locations in the extended terminal airspace area, 2) the optimal merge locations tend to be positioned where large

uncertainties are present so that pilot's control efforts reduce the local uncertainties at the control point.

The rest of this paper is organized as follows: First, the basic formulation of the original MILP algorithms is explained, and the objectives and constraints for our route assignment and scheduling problem are defined. Second, a number of heuristics are introduced into the original MILP formulation as ways to reduce computational cost. Third, the dynamic planner framework with uncertainty modeling and extra separation buffers are detailed. Finally, a design of optimal route structure in terms of the merge topologies is discussed and followed by conclusions and future work.

## Problem Formulation

We apply a MILP-based formulation to the scheduling problem in the extended terminal airspace area, from the entry fixes to runway, and only the arrival portion of the scheduling will be considered. The plan consists of a time-constrained route for each aircraft in the demand set, with STA specifications at control points along each route, as dictated by separation rules, such that the resulting movement plan for all aircraft in the demand set is conflict-free.

### *Basic Mixed Integer Linear Programming*

A MILP scheduler has advantages in the scheduling problems over the traditional FCFS scheduler [9]. The design variables can be freely chosen specifically to a problem as forms of binary variables and continuous variables. However, the computational cost of a branch and bound search algorithm is expensive and its scalability with the problem size is rather poor and a number of heuristics to reduce the computational burden will be explained in later sections.

### *Initial Demand Set*

Let  $\mathcal{F}$ ,  $\mathcal{R}$ , and  $\mathcal{P}$  define a set of flights, available routes, and scheduling points, respectively, and  $N_F$ ,  $N_R$ , and  $N_P$  are the total number of flights,

routes, and scheduling points in the demand set.

$$\begin{aligned}\mathcal{F} &= \{f_j \mid 1 \leq j \leq N_F\} = \{f_1, f_2, \dots, f_{N_F}\} \\ \mathcal{R} &= \{r_j \mid 1 \leq j \leq N_R\} = \{r_1, r_2, \dots, r_{N_R}\} \\ \mathcal{P} &= \{p_j \mid 1 \leq j \leq N_P\} = \{p_1, p_2, \dots, p_{N_P}\}\end{aligned}$$

We further presume the existence of functions that select the subset of routes from  $\mathcal{R}$  that are feasible for a given flight  $f \in \mathcal{F}$  and the subset of points from  $\mathcal{P}$  that are feasible for a given route  $r \in \mathcal{R}$ .

$$\mathcal{R}^f = \{r_k \mid \text{all possible routes that flight } f \text{ can fly, } k \in \{1, 2, \dots, N_R(f)\}\} \quad (1)$$

$$\mathcal{P}^r = \{p_k \mid \text{ordered set of scheduling points on the route } r, k \in \{1, 2, \dots, N_P(r)\}\} \quad (2)$$

where  $N_R(f)$  is the number of all routes that flight  $f$  can fly, and  $N_P(r)$  is the number of all the scheduling points on route  $r$ . Then, it can be easily inferred that a set of total routes and the points are the superset of  $\mathcal{R}^f$  and  $\mathcal{P}^r$ , respectively, and the relation is represented as follows:

$$\mathcal{R} = \bigcup_{f \in \mathcal{F}} \mathcal{R}^f \text{ and } \mathcal{P} = \bigcup_{r \in \mathcal{R}} \mathcal{P}^r$$

### *Decision Variables*

Given the notation in the previous section, the decision variables of interest for this problem can be defined:

- $A_{f,r}$  - A binary route assignment variable that takes on the value of 1 if flight  $f$  is assigned to the route  $r$  and zero otherwise.
- $T_{f,r,p}$  - A continuous variable representing the time that flight  $f$  is scheduled to cross the scheduling point  $p$  on route  $r$ , where  $f \in \mathcal{F}$ ,  $r \in \mathcal{R}^f$ , and  $p \in \mathcal{P}^r$ .
- $S_{f,f',r,r',p}$  - A binary variable that takes on the value of 1 if flight  $f$  on route  $r$  is sequenced prior to flight  $f'$  on route  $r'$  at shared scheduled point  $p$ , where  $f \in \mathcal{F}$ ,  $r \in \mathcal{R}^f$ ,  $r' \in \mathcal{R}^{f'}$ , and  $p \in \mathcal{P}^r \cap \mathcal{P}^{r'} \neq \emptyset$ .

## Objective and Constraints

For the purposes of this paper, the objective function is defined so as to minimize the total time required for all flights to reach the end of their route, i.e., the runway threshold:

$$J = \sum_{f \in \mathcal{F}} \sum_{r \in \mathcal{R}^f} A_{f,r} T_{f,r,p_F} \quad (3)$$

The problem constraints are as follows:

- *Assignment to Only One Route.* Each flight can be assigned to one and only one route.
- *Crossing Time at Initial and Final Point.* If a flight is assigned to a given route, then its start time on that route must be no earlier than the earliest feasible time on that route,  $T_{f,r}^E$ , and no later than the latest feasible time,  $T_{f,r}^F$ .
- *Ordering Constraint at Potentially Shared Scheduling Points.* For each pair of flights and each pair of route assignment options that share a common scheduling point, the order in which the flights are sequenced at the common point must be uniquely specified.
- *Separation Constraint.* At each common scheduling point, successive aircraft must be separated by a minimum time that is potentially intersection-dependent.
- *Transit Time Constraints.* In order to be physically realizable, the travel time between scheduling points must be greater than the minimum possible travel time and should be bounded by the maximum “delayability” of the flight between the scheduling points.

## Heuristics-Based Schedulers

One of the drawbacks of the MILP-based scheduler is its prohibitive computational cost. However, observation of the branch and bound search procedure indicates a large portion of search time is spent on evaluating unrealistic combinations of route assignment and sequencing. Heuristics in the route assignment and the queuing can help eliminate some of the unrealistic search efforts and

reduce the computational burden. In our study, we tried two types of heuristics. First, heuristics based on the FCFS scheduling behavior can eliminate a number of binary decision variables in the original MILP formulation. Second, the branch and bound search algorithm can be replaced by GA-based heuristics.

### *Heuristics-Based Mixed Integer Linear Programming*

Based on the observations of the FCFS scheduling strategy, assumptions are made on sequencing along certain route segments, such as merge portions. The details of the following heuristics were explained in our previous work on the metroplex operations and only brief summaries are listed here:

- *Precedence constraint heuristic.* Sequences along the common segments of the routes do not change and follow the queuing from the upstream of the merging segments.
- *Windowing heuristics.* A sequence change is not allowed for a pair with their earliest crossing times at the entry fixes separated by more than a certain amount, i.e., windowing value. Using windowing heuristics, resulting schedules are planned locally and subsequently corresponding to this value.
- *FCFS heuristics.* The ordering at all scheduling points can be predetermined based on unimpeded transit times to a specific scheduling point, a runway or an entry fix. This strategy is equivalent to the FCFS scheduling ideas and the computational cost benefits are maximum.

A scheduling performance almost equivalent to that of the original MILP formulation was obtained at only a fraction of the computation cost of the original MILP formulation. However, the cost for the branch and bound search is still expensive though not as prohibitive as in the original MILP formulation.

### *Heuristics of Genetic Algorithms*

A GA is a stochastic search method widely used in numerous optimization areas and has ad-

vantages in handling both discrete and continuous design variables [10]. GAs are simple in mathematical formulation but are typically expensive in computation due to their stochastic search procedure.

The idea of adopting GAs in the scheduling problem for determining the binary decision variables in lieu of the branch and bound searches has been suggested and used in surface management work [7] and metroplex operations [8].

Once the binary variables of the route assignment and the sequencing are determined through GAs, computation of the STAs is carried out by an LP solver. Since the LP is very efficient in computation and takes less than a couple of seconds to schedule hundreds of flights, the idea of combining GAs with pure LP procedure is favored for the dense terminal airspace simulations. Advantages of the GA-based MILP planner include:

- It allows for the solution to be seeded with a good initial guess, based on heuristics.
- All individual candidate solutions are, by definition, feasible solutions - thus a usable solution is available at all times.
- The solution tends to improve with computational time.
- It naturally handles windowing heuristics.

### Individual Candidate Solution Representation

Each individual candidate solution, or “an individual” in short, consists of two vectors: an assignment vector and a sequence vector. The length of each vector is equivalent to the number of flights in the demand set. Each element of the assignment vector represents a possible route assignment for a flight in the demand set. Given the structure of the route, a sequence of the scheduling points in the route is prescribed, and the sequence vector is defined at each scheduling point as a possible sequence of the flights passing through that point. Route assignments and the queuing are random based on the stochastic nature of the GAs. A constraint of no passing in the sequence vector along the common route segment is enforced to further reduce the computational cost.

### Mutation

To maintain the diversity of the individual from one generation to the next, mutations are carried out at a specified probability in each generation to both vectors of route assignment and sequence. Due to the coupling between the assignment map and sequence map held within each individual, the mutation operators are applied sequentially. The assignment mutation simply consists of randomly replacing the route assigned for an individual with another value from the feasible set of routes for the flight. Then, the sequence map for each individual is updated based on the mutated route assignment map. A sequence mutation is applied to each scheduling point at a given route assignment. Swapping sequence is determined via probability. If a random number sample is less than the specified probability of mutating sequence, then the number of swaps are chosen from a  $(0, N_{\max})$ , where  $N_{\max}$  is the maximum number of swaps. A random pair of flights that contain this scheduling point in their assigned routes swap their ordering relative to the current ordering.

### Fitness and Selection

Fitness of each individual is defined as an objective function value and evaluated by solving the pure LP problem implied by its route assignment and sequence. The definition of the objectives and the imposition of the constraints are equivalent to those of the MILP formulation: earliest transit time limit, lower and upper bounds of the transit time via the specified speed controllability, and separation requirements specific to the aircraft type at the scheduling points.

Once each individual in the population has been assigned a fitness, the selection of individuals to form the basis of the next generation is performed using a simple tournament selection scheme. A tournament scheme finds the best-performing  $P/2$  number of individuals, where  $P$  is the size of the population, and those are selected as the basis of the next generation. These  $P/2$  number of individuals are then mutated to form the population of size  $P$  to be evaluated. This cycle of fitness-selection-mutation is repeated until a specified number of generations are completed.

## Comparison of Computation Time and Optimality

The optimality and the computation times are compared for the schedulers that were described in previous sections. The terminal route structure tested for comparisons is a binary route topology with double merges: 4 route options and 8 scheduling points (4 entry fixes, 2 merge points and 1 runway). The route topology is shown in Figure 1 with the entry fixes of WP<sub>31</sub> through WP<sub>34</sub> and the runway of WP<sub>0</sub>. The number of flights varies from 6 to 100. Computation times with respect to varying number of flights and the average delays of 8 flights are compared in Table 1. The expensive computational costs of the original MILP scheduler and the FCFS heuristics-based MILP scheduler prohibited computations for more than 8 flights and 40 flights, respectively and their values are represented as N/A in Table 1. Although the FCFS heuristics allows scheduling up to 40 flights, the computation time is not still satisfactory for the dense terminal airspace simulation. However, the speed-up of the computation time for the GA-based MILP planner is considerable even with 100 flights. It is observed from the additional scheduling of a larger number of flights using GA-based MILP planner that it is able to handle hundreds of flights in a few minutes. Therefore, it is concluded that the GA-based MILP scheduler is faster than the others and is more appropriate for the scheduling problem in the dense terminal airspace operation.

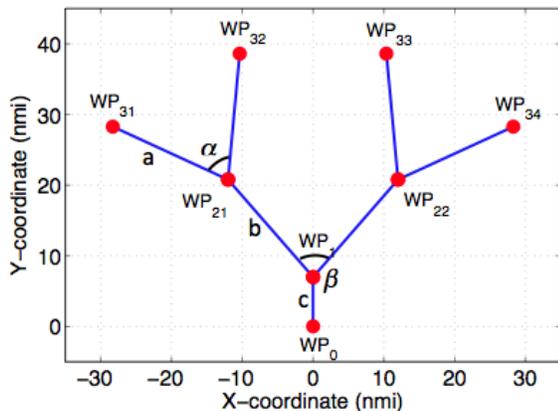


Figure 1. Example route topology

Table 1. Computation times (in seconds) to schedule different number of flights

Number of flights	6	7	8	40	100	Average delay (8 flights)
MILP	6.7	98.3	1258.4	N/A	N/A	8.23
MILP + FCFS heuristics	2.1	33.2	419.3	2040	N/A	8.38
MILP + GA heuristics	$\leq 1.0$	1.0	2.2	23.4	50.0	8.58

## Dynamic Planner Framework

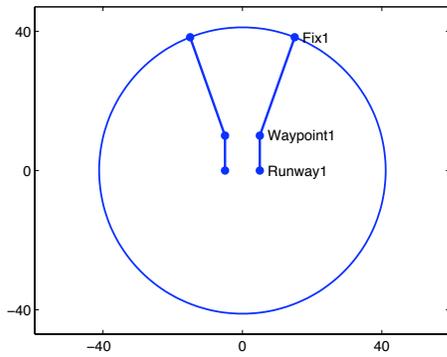
The schedulers described in the previous section are deterministic. The transit time of any route segment was a function of aircraft type and the speed profile only, and the uncertainties along the routes were not considered in the planning. Even if the uncertainties are taken into account in the planner, the aforementioned schedulers are based on the premise that the uncertainties of each aircraft are known prior to the planning along any route segments. However, the STAs in the real-time simulation should be updated in a dynamic manner corresponding to the varying situations of weather, wind and off-nominal scenarios. A key to the realistic scheduling is a dynamic update of the STAs in a real-time trajectory model in consideration of the uncertainties in flight simulation.

A dynamic planner framework is developed in our study by interactively integrating the trajectory simulation and the schedule planning for STA update. The framework consists of two components:

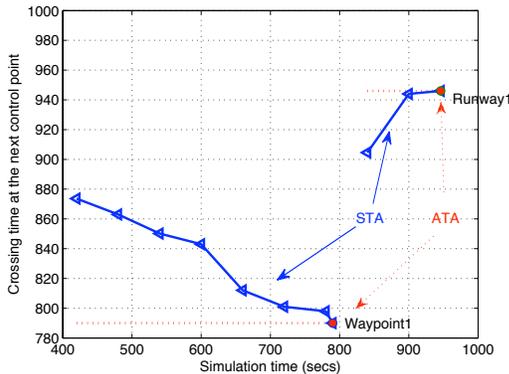
- **Simulator:** This module is responsible for advancing time. It manages the creation of targets at specified location and time, and constructs the demand snapshot at a given instance of time. The module delegates to a trajectory model that handles the actual movement of flights along their most recent plan and blends motion between successive plans. Uncertainties in the transit time along the route segment and their propagations are implemented in the simulation module, which will be explained in later section.
- **Planner:** This module is responsible for con-

structuring conflict-free plans for each aircraft in a given demand set. Each motion plan consists of a sequence of waypoints with an associated STA. Although any type of the scheduler can be used in the dynamic planner framework, the GA-based MILP scheduler is integrated into the current framework. Furthermore, to take into account the uncertainties in the trajectory simulation, an extra buffer additional to the desired separation is added to mitigate effect of uncertainties so that the resulting schedule remains conflict-free. The details of the additional buffer are discussed in later section.

A specified amount of controllability is allowed in speed profile to maximize the scheduling performance and is implemented in both the planner and the trajectory model simulator.



(a) Route topology



(b) Trajectories of STAs and Actual Time of Arrival (ATA): STAs in solid lines and ATA as the last point of the line.

**Figure 2. Example of dynamic planner framework**

## Trajectory Model Simulation

Flight simulation of the trajectory model is made via subsequent communications with the planner. First, the dynamic planner starts from the pre-planning of the initial demand set. Given the speed profile and the ETA of each aircraft to the first schedule point, the initial STAs are computed by the scheduler at all scheduling points on the assigned route. The STAs are predicted such that they satisfy the constraints of the transit time bounds on the route segments and the desired separation at the scheduling points. The trajectory model periodically computes the distance from the current position to the next scheduling point. For our work, the update period is 60 seconds. With distance to the next scheduling point and the STA predicted by the planner, the target speed is calculated and checked whether it is bounded by the speed range specified in the original speed profile. Once the target speed is determined, then the trajectory simulator advances the flights by an update period. After the simulation, the aircraft position is recalculated and the earliest time to the scheduling point is updated. A subsequent planning cycle updates the STAs based on the most recent simulation results. This cycle of simulation and planning is iterated and advanced in time by the update period until all the flights arrive at the runway. An example of the cycles of simulation and planning is shown in Figure 2. Given a simple route structure shown in Figure 2(a), the STAs at the scheduling points of "Waypoint1" and "Runway1" are updated at each update period of 60 seconds. Their convergence history is plotted in Figure 2(b), and STA updates are shown by the triangles. Unlike the static planner, the STA values are updated at each update period and finally coincide with the Actual Time of Arrival (ATA) values at the scheduling points. The speeds along the routes change correspondingly in each update period.

## Controllability

To delay or expedite an aircraft on its way to the next scheduling point, the controllability on any route segment is modeled such that it allows  $\pm 10\%$  speed variation. The corresponding transit time bounds are computed. This controllability

is derived from the statistics of the aircraft flying with modern avionics and the onboard precision system [11].

### ***Uncertainty Modeling and Propagation***

Accurate prediction of uncertainties along an entire aircraft’s trajectory is not trivial. It is a complicated function of space and time, which requires precise understanding of where and how much of the uncertainties are present and how they affect individual aircraft operations. However, the uncertainties in the runway arrival times are quantifiable from the statistics of the runway arrival times observed in a given duration of time. We can model the aircraft arrival time prediction errors at the runway by a normal distribution with the time-invariant mean and standard deviation values. On the other hand, the uncertainty at the intermediate control points and route-merge points require a mathematical model of the uncertainty propagation mechanism along the routes. A simple linearized form is introduced in our simulation module: variance at each scheduling point is assumed to be proportional to the variance of the runway arrival time prediction errors when there are no control effort in between the scheduling point and runway. Uncertainty amount at any point on the route is scaled by the ratio of the intermediate route segment length to the entire route length from the entry fix to the runway. This presumes that the uncertainties grow longer along the longer routes since the flight is associated with longer transit time without control efforts.

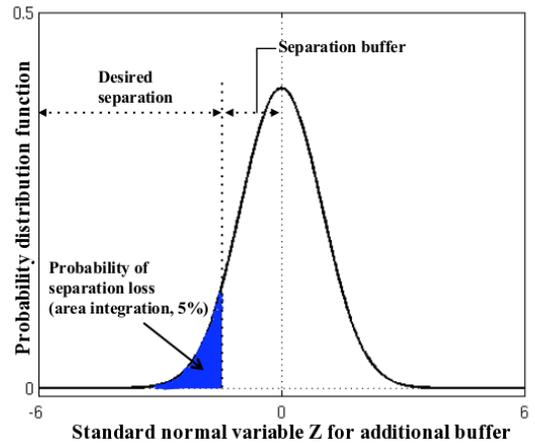
Based on the central limit theorem, we assume that the position error of aircraft at the scheduling point is approximately normally distributed. Then, the corresponding inter-arrival error of any pair is also normally distributed, and the mean and variance of inter-arrival errors are estimated by the following basic relationships: if  $X$  and  $Y$  are independent random variables that are normally distributed, then  $X + Y$  is also normally distributed, i.e., if  $X \sim N(\mu, \sigma^2)$  and  $Y \sim N(\nu, \tau^2)$  and  $X$  and  $Y$  are independent, then  $aX + b \sim N(a\mu + b, a^2\sigma^2)$  and  $X + Y \sim N(\mu + \nu, \sigma^2 + \tau^2)$ . The means and the variances of the  $X$  and  $Y$  are the  $\mu$  and  $\nu$ , and  $\sigma$  and  $\tau$ , respectively.

The validity of the above relations holds best when the independence of two variables,  $X$  and  $Y$ ,

is relatively well guaranteed. The inter-arrival error of a pair of aircraft is a complicated function of many factors such as precision errors in navigation and weather including wind. We assume for simplicity in our trajectory simulation that the wind effect during the traffic simulation is rather constant. Direction and magnitude of the wind are relatively non-changing on each aircraft throughout the whole simulation, then we can treat the wind effect as a constant that is freely addable / subtractable to/from the standard deviation of the position error of each aircraft at all scheduling points, and the above relation holds relatively well.

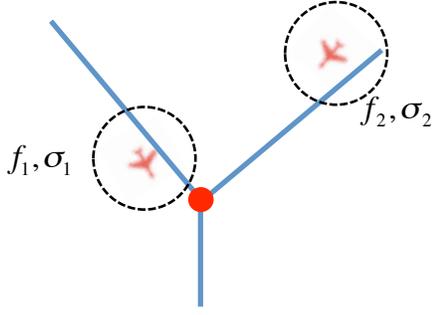
### ***Additional Separation Buffer due to Uncertainties***

Aircraft arrival time errors and the corresponding inter-arrival error in a pair are likely to cause violations of the desired separation. In order to ensure desired separation is maintained in spite of arrival time error, extra buffers are added to the original desired separation [12]. The amount of the additional buffers is determined from the probability of the inter-arrival errors and its normal distribution shown in Figure 3. If we choose a value,



**Figure 3. Probability distribution of position errors**

$Z$ , for an additional buffer such that the cumulative probability corresponding to  $Z$  coincides with a specified confidence level, 90% for the current work, then we can say that the separation requirement in any pair will be satisfied under uncertainty with 90% confidence and be violated with 10% tol-



**Figure 4. Uncertainties of position errors in a pair at the merge point ( $\sigma_1 = \sigma_{\text{leader}}$  and  $\sigma_2 = \sigma_{\text{follower}}$ .)**

erance. A brief graphical explanation is shown in Figure 3. This sets the additional buffer value as  $1.645\sigma$ , where  $\sigma$  is the standard deviation of the inter-arrival error distribution.

The above can be expressed mathematically as follows. Standard deviations of the arrival time errors of a leader and a follower in a given pair are denoted as  $\sigma_{\text{leader}}$  and  $\sigma_{\text{follower}}$ , respectively as in Figure 4. The amounts are scaled from runway standard deviation in proportion to the ratio of the local route segment to the entire route from runway. Assuming the probability of the position errors of a leader and a follower are independent of each other, the standard deviation of the inter-arrival error is assumed to be  $\sqrt{\sigma_{\text{leader}} + \sigma_{\text{follower}}}$ . A corresponding additional buffer is set as  $1.645\sqrt{\sigma_{\text{leader}}^2 + \sigma_{\text{follower}}^2}$  for a 90 % confidence interval based on the normal distribution of the inter-arrival errors. A simple mathematical formulation of total amount of buffers is expressed in following Equation.

$$\begin{aligned} t_{\text{sep\_tot}} &= t_{\text{sep\_desired}} + t_{\text{sep\_}\sigma} \\ &= \frac{d_{\text{sep\_desired}}}{V} + 1.645\sqrt{\sigma_{\text{leader}}^2 + \sigma_{\text{follower}}^2}, \end{aligned}$$

where  $V$  is airspeed and  $t_{\text{sep\_tot}}$  represents total amount of separation requirement. Terms of  $t_{\text{sep\_desired}}$ ,  $d_{\text{sep\_desired}}$  and  $t_{\text{sep\_}\sigma}$  represent a desired separation in time, a desired separation in distance, and an uncertainty-related, additional buffer, respectively.

## Application: Optimal Route Structure Under Uncertainty

A practical application of the developed schedulers and the dynamic planner framework is shown in this section. A design of an optimal route structure under uncertainty in the extended terminal airspace area is carried out to improve scheduling performance and, thus, to best utilize the limited airspace resources.

First, key parameters for a route structure design are identified. A total of five example route structures are constructed with varying design parameters. A static FCFS heuristics-based MILP planner is used to analyze the scheduling performance of each notional route structure, but the results are validated by using the dynamic planner framework. Based on the analyses of the scheduling performance of the notional route structures, more general cases of various merge topologies are considered subsequently.

### Parameterization

A route structure consists of such parameters as the location and number of entry fixes, runways and merge points as well as route segment lengths. A demand set is also critical in scheduling performance. The demand set defines relevant flight information including the total number of flights, arrival time at entry to the route structure, and traffic duration time. In fact, the scheduling performance is very sensitive to the demand set. A fully saturated demand set, i.e., one which has no periods of low demands, is used to isolate the scheduling performance from the effects of the route structure alone. A fully saturated traffic flow is consistent with dense terminal airspace operation whereby demand exceeds capacity for extended periods of time. In this way, the runway capacity is always exhausted and the number of runways becomes no longer a parameter of the airspace topology. An entry fix topology, i.e., the total number of entry fixes and their locations, is also assumed to be given to facilitate the fully saturated traffic flow. Thus, the main parameter in our design study is the merge point topology, i.e., the location and number of the merge points.

## Numerical Test I

First, a numerical experiment is performed on five route structures having different topologies with varying numbers and locations of the merge points. Figures 5 and 6 have a single merge point whereas Figures 7 through 9 have two merge points. The locations of the merge points are moved in order to vary the ratios of route segment lengths in a given topology and thus vary the uncertainty distribution along the route. These topologies can also be defined by the parameters of the route segment length and the merge angle between two routes. For example, Figure 7 through 9 can be defined by varying the parameters of  $a$ ,  $b$ ,  $c$ ,  $\alpha$ , and  $\beta$  as shown in Figure 1, where  $a$ ,  $b$  and  $c$  represent the route segment length, and  $\alpha$  and  $\beta$  represent angles between two merging routes.

A FCFS heuristics-based MILP is used for computing the scheduling performance of each topology. In the cost comparison of the schedulers explained in previous sections, a demand set of 80 flights is not trivial in scheduling even for the heuristics-based MILP scheduler. Total computation time grows very quickly, especially for a stochastic case where hundreds or thousands of Monte Carlo simulations are performed. Thus, for this numerical experiment, the route is pre-assigned for each aircraft and the orders at the merge points are predetermined based on the the unimpeded transit times from the entry fix to the runway. For a stochastic scenario, an uncertainty model is directly implemented in the scheduler, and we do not employ a dynamic planner for this preliminary numerical experiment. However, a more realistic validation of these five airspace topologies is carried out by the dynamic planner and analysis results will be shown in the following section.

Initially, a total of 80 flights pass through the entry fixes equally divided into four streams and follow their pre-assigned routes during a short period of 100 seconds. This short duration time ensures fully saturated air traffic. Four types of weight class categories are used: “heavy”, “B757”, “large”, and “small”. A majority of aircraft in our demand set, more than 90%, belong to the “large” type, indicative of today’s operations.

The total amount of separation required at each scheduling point is the sum of the minimum desired separation and the extra additional buffer to

Table 2. Desired separation (nmi)

leader \ follower	heavy	B757	large	small
heavy	4	5	5	5
B757	3	3	3	4
large	3	3	3	4
small	3	3	3	3

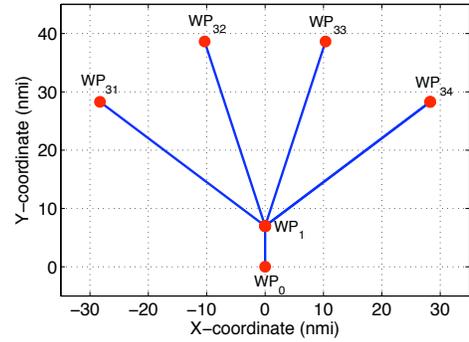
mitigate uncertainty. The quantification of this amount is done using the formulation of Equation 4. The amount of the desired separation is based on the weight classes of the leader and the follower in a given pair, and the values are specified in Table 2. Airspeed is assumed to be linearly decreasing along the routes in the extended terminal airspace area and its value at each scheduling point is summarized at the tables in Figure 5 through 9. Airspeeds at the entry fixes are fixed at 250 kts. Additionally, the desired separation of the entry fix was 5 nmi to model the separation required for en route airspace. Previously collected data indicates that a standard deviation of uncertainty (in the aircraft arrival time errors) at the entry fix is found to be approximately 30 seconds. The additional uncertainty-related separation buffer required at the entry fix in order to mitigate this uncertainty is calculated in a similar way for the merge points and runway.

Given the prescribed uncertainty and its corresponding separation buffer, the FCFS heuristics-based MILP scheduler simulates both the deterministic and stochastic scenarios for all airspace topologies shown in Figure 5 through 9. A total of 500 Monte Carlo simulations were carried out for the stochastic simulations. The separation buffer and total separation are computed for each case and summarized in the tables in Figure 5 through 9. The resulting average delays are shown in Table 3. The average delay is calculated as a difference between unimpeded transit time and actual transit time. The value of average delay is an artifact of the fully saturated traffic scenario. The key result is the performance improvement.

First, for the deterministic scenarios where no consideration of uncertainty is made, the third column in Table 3 implies that the scheduling performance is relatively independent of a particular airspace topology and its merge point locations. Although a slight improvement is shown in

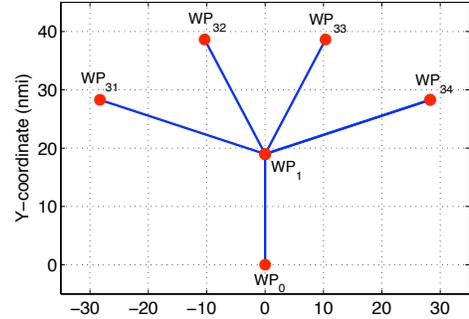
waypoint	WP <sub>21</sub>	WP <sub>1</sub>	WP <sub>0</sub>
air speed (kts)		160	130
t <sub>sep_desired</sub> (sec)		67.5	83.1
t <sub>sep_σ</sub>		29.13	6.063
t <sub>sep_tot</sub>		96.63	89.14
average spacing		99.33	99.43

Figure 5. Case 1: single late merge.



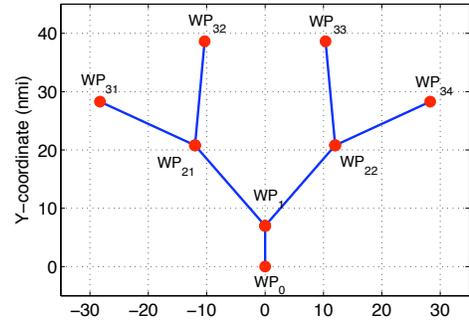
waypoint	WP <sub>21</sub>	WP <sub>1</sub>	WP <sub>0</sub>
air speed		200	130
t <sub>sep_desired</sub>		54	83.1
t <sub>sep_σ</sub>		21.3	17.0
t <sub>sep_tot</sub>		75.3	99.2
average spacing		100.48	102.10

Figure 6. Case 2: single early merge.



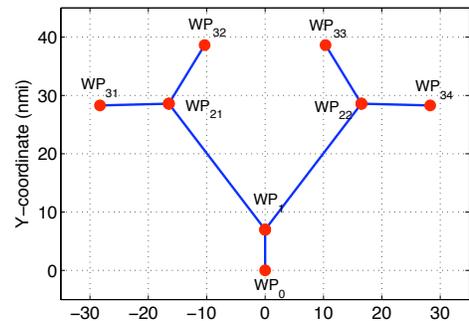
waypoint	WP <sub>21</sub>	WP <sub>1</sub>	WP <sub>0</sub>
air speed	200	160	130
t <sub>sep_desired</sub>	54	67.5	83.077
t <sub>sep_σ</sub>	14.8	14.76	5.653
t <sub>sep_tot</sub>	68.48	82.26	88.73
average spacing		92.67	92.91

Figure 7. Case 3: double merges, equally distanced.



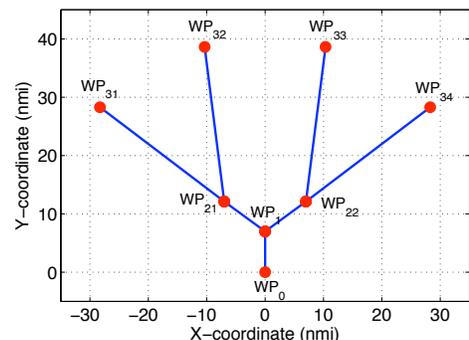
waypoint	WP <sub>21</sub>	WP <sub>1</sub>	WP <sub>0</sub>
air speed	225	160	130
t <sub>sep_desired</sub>	48	67.5	83.077
t <sub>sep_σ</sub>	8.968	20.64	5.32
t <sub>sep_tot</sub>	56.97	88.142	88.397
average spacing		92.61	92.66

Figure 8. Case 4: double early merges.



waypoint	WP <sub>21</sub>	WP <sub>1</sub>	WP <sub>0</sub>
air speed	180	160	130
t <sub>sep_desired</sub>	60	67.5	83.077
t <sub>sep_σ</sub>	16.157	7.144	5.76
t <sub>sep_tot</sub>	76.157	74.64	88.837
average spacing		92.88	93.02

Figure 9. Case 5: double late merges.



**Table 3. Predicted average delays (seconds).**

Topology	Cases	Average delay (Deterministic)	Performance improvement (w.r.t Case 1)	Average delay (Stochastic)	Performance improvement (w.r.t Case 1)
single	Case 1	3363.4	1.0 %	3834.8	1.0 %
merge	Case 2	3370.8	-0.2 %	3918.2	-2.1 %
double	Case 3	3348.1	0.45 %	3571.6	6.8 %
merge	Case 4	3343.6	0.6 %	3564.8	7.0 %
	Case 5	3352.0	0.3 %	3579.7	6.7 %

the double merge cases, differences in the average delays among all cases are very minor, less than 1%. This can be explained from the formulation in Equation 4, where the amount of total separation is solely a function of airspeed alone when there is no uncertainty. As airspeed gradually decreases towards the runway, the runway always requires the biggest separation of all the scheduling points. This makes the scheduling performance largely insensitive to the particular upstream route structure. The controllability of each aircraft does not affect the scheduling performance either, as most of the aircraft have to absorb delays propagated from the preceding aircraft.

Second, for the stochastic scenarios, an improvement of 6.7 % is shown in Case 5, when comparing the single merge and the double merge topologies. Uncertainty creates perturbations in the transit times, and some of the aircraft can exploit their controllability to fill the gaps in a pair created by the uncertainty.

The results from both deterministic and stochastic cases in Figure 5 and 6 are interesting and informative. The actual, average spacings between all pairs are extracted from the scheduling results for both cases and shown in blue in tables in Figure 5 and 6. A careful comparison of five cases of the resultant average spacings and the required separation indicates that the average spacing at each scheduling point is dominated by the largest separation required along the route. This is shown in red in tables in Figure 5 and 6. This observation suggests that in a fully saturated traffic flow, the traffic becomes steady with acceleration and deceleration adjusted by the controllability that is allowed in each route segment and incurs approximately the same amount of spacings in any pairs.

### *Validation Using Dynamic Planner*

The scheduling results in Section of Numerical Test I are validated using the dynamic planner framework. A GA-based MILP is employed for the planner of the framework for a comparison with the FCFS heuristics-based MILP. Unlike the static planner used in Section of Numerical Test I, a dynamic planner involves iterative interactions between the planner and the simulator as time advances. An update period of 60 seconds was used in our validation. In each dynamic planning cycle, the simulator tries to track the STAs provided by the planner at all scheduling points, and the planner creates new schedules as a result of updated aircraft positions and ETAs from the simulator. A realistic demand set is critical in the dynamic planner so that the trajectory model can be simulated based on the operationally reasonable and realistic schedule plans. For this reason, rather than using fully saturated traffic as in the Numerical Test I, initially well-separated traffic flows, by 5 nmi in any pairs with random deviation ranging from -20% to +20%, are used for the dynamic planner. This results in hour-long traffic flows for the same number of flights. Also the initial departing point for all aircraft is fixed at about 100 nmi away from the runway and is almost aligned with the freeze horizon.

The validation results are summarized in Table 4. An average delay value is, again, defined as a difference between the unimpeded transit time and the actual transit time from the entry fix to the runway. Compared to the values in Table 3 which used a fully saturated traffic flow, average delay values become more reasonable with the maximum value less than five minutes.

It should be noted that the dynamic planner is more computation-intensive than the static scheduling planner, as the specified plan update period cannot be set too large in a real-time simulation. A wall-clock CPU time per dynamic planning takes about an hour with 60 seconds update period. The dynamic planning requires hundreds of cycles of simulation and planning for aircraft to travel the entire route of 100 nmi in length. Thus, the computation time of the dynamic planner for the stochastic case becomes very expensive with Monte Carlo simulation. The results for the stochastic cases shown in Table 4 are obtained from only 100 Monte Carlo simulations. More simulations are planned as part of the future work.

It can be concluded from the average delay results shown in Table 4 that Case 5 is the best-performing airspace topology and it has a performance improvement, compared to Case 1, of approximately 50% in the deterministic case and approximately 30% in the stochastic scenario. Although the sample size of the Monte Carlo simulations in the stochastic scenarios is not big enough to make the conclusion more trustworthy, the standard deviation of the 100 Monte Carlo samples are as little as 15 seconds resulting in 5% tolerance that is far less than our percentile performance improvement. It should also be noted that for a less saturated traffic flow, an improvement in the scheduling performance is more dramatic compared to the 6.7% in Table 3. This can be explained by the fact that in fully saturated traffic flow, all the aircraft quickly exhaust their controllability and are assigned their slowest speed profile. That is not the case with less saturated air traffic, and the benefits from the controllability are more dramatic in this case. It should also be noted that the same trend of the static planning in Numerical Test I is shown in the validation results: an airspace topology with small separation requirements at the scheduling points is more favorable in the scheduling performance.

## *Numerical Test II*

Based on the results in the previous sections that the scheduling performance is heavily dependent on the amount of maximum separation at the scheduling points, a simple numerical experiment is carried out to investigate more general variations

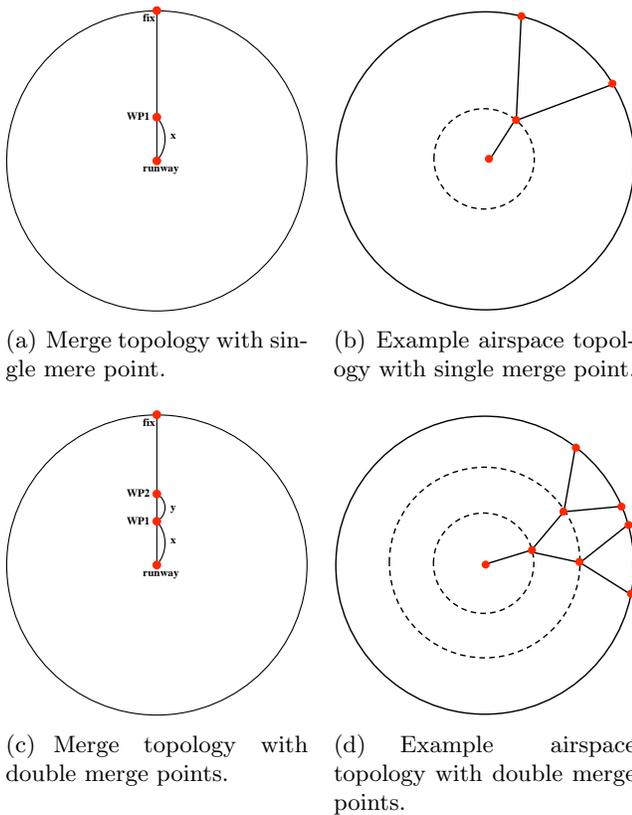
of the airspace topology. Relationships among the component separation amounts at the scheduling points are analyzed when the merge points move around in the extended terminal airspace area. The airspace topology is simplified to a circle with 40 nmi radius as shown in Figure 10. The points, WP1 and WP2, represent the merge locations that can move circumferentially at a relatively constant radial distance away from the runway, and the routes can be merged at these locations. Radial distances,  $x$  and  $x + y$ , are also allowed to vary within radius bounds:  $0 < x < R$  and  $0 < y < R - x$ , where  $R$  is the radial distance from the runway to the entry fix. The movement of the merge points, WP1 and WP2, are shown in Figures 10(a) and 10(c), and the corresponding example airspace topologies are plotted in Figures 10(b) and 10(d), respectively. As WP1 and WP2 move about in the extended terminal airspace area, the entire route length in a particular airspace topology from the entry fix to the runway and the corresponding transit time may change. However, any large variations were excluded in the transit times by locating the entry fix such that the entire route from the runway to the entry fix does not deviate much from the original radial lines.

For the airspace topology with a single merge as in Figures 10(a) and 10(b), total separation amounts at the runway and the WP1, are computed from Equation 4 and plotted as solid lines with symbols in Figure 11(a). The two dashed lines represent the desired separation amount using the nominal speed profile, and the uncertainty-related, extra separation buffers are plotted as dotted lines. The traces of the maximum separation with respect to the varying merge locations are shown in Figure 11(b). Figure 11(b) implies that there exists an optimal merge location somewhere around 5 or 6 nmi away from the runway.

The case of double merges inside the extended airspace area as in Figures 10(c) and 10(d) shows similar trend as the single merge case. At the given downstream merge point location,  $x$ , which can move from the runway to the entry fix, the upstream merge location,  $y$ , also traverses between downstream merge point and the entry fix. The plot of the maximum separation buffer of all scheduling points, i.e., runway, downstream and upstream merge points, are almost identical to the plot of Figure 11(b) and is omitted in the paper; how-

**Table 4. Average delays (seconds) validated by dynamic planner framework for stochastic case.**

Topology	Cases	Update cycles	Deterministic scenario	Performance improvement (w.r.t. Case 1)	Update cycles	Stochastic scenario	Performance improvement (w.r.t. Case 1)
single	Case 1	304	249.1	1.0 %	350	291.0	1.0 %
merge	Case 2	257	235.0	5.6 %	329	283.0	2.7 %
double	Case 3	228	107.26	56.9 %	294	228.2	21.6 %
merge	Case 4	232	108.31	56.5 %	295	213.5	26.6 %
	Case 5	247	109.17	56.0%	294	199.3	31.0 %



**Figure 10. Simplified airspace topology of the extended terminal area with varying merge point locations**

ever, this indicates that the separation at the runway typically requires the biggest amount and in other cases the downstream merge point requires a larger buffer than the upstream merge point.

The exact optimal locations of merge points are less meaningful in our analysis, as they can vary depending on the underlying airspeed profile, uncertainty quantification and propagation model, and the assumed additional separation buffer due to uncertainty. Possible changes to the optimal merge locations can be inferred from Figure 11(a).

Desired separation represented as the dashed lines in blue and red are rather non-changing as these are solely determined by the airspeed topology alone, and the assumption of monotonically decreasing airspeed in the extended terminal airspace area is reasonable. On the other hand, the modeling of uncertainty and its propagation in the extended terminal airspace area is still an area of active research that requires extensive studies on how the uncertainties are distributed or propagated along the routes. How temporal or spatial deviation from its predicted trajectory behavior is quantified over time and distance and translated into the time-based scheduler is a difficult and yet very important topic in the scheduling and real-time simulations.

### ***Various Uncertainty Propagation Models***

A brief sensitivity study of the optimal merge locations is carried out with respect to the different models of uncertainty distribution and propagation. The results in previous sections assumed that the increment of the uncertainty per unit route segment is constant and the resultant uncertainty

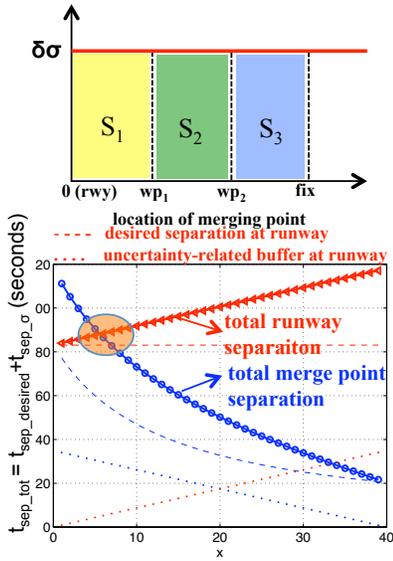


Figure 12. Constant distribution of uncertainty

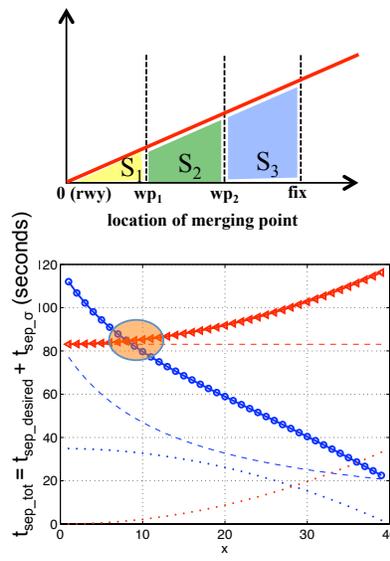


Figure 13. Linear distribution of uncertainty

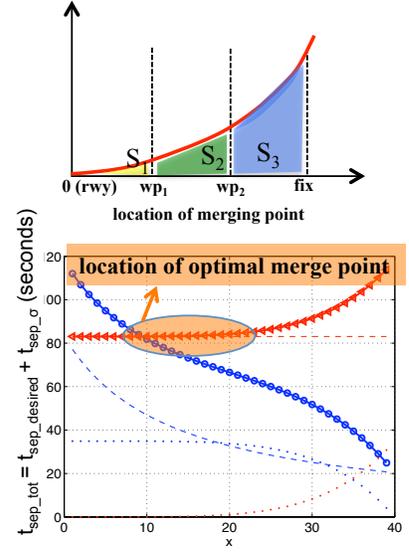
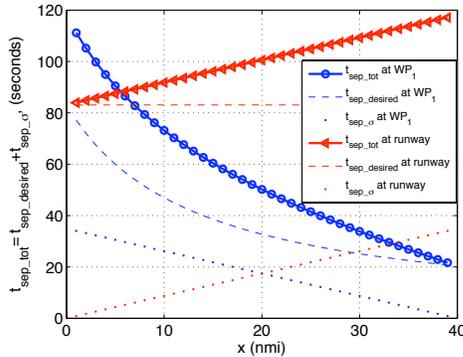
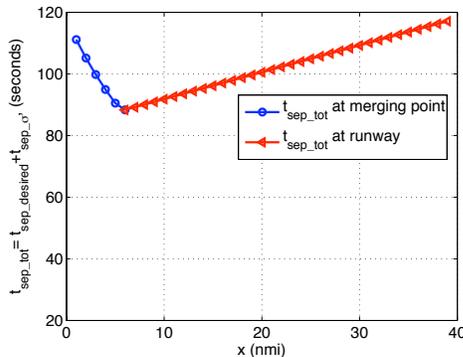


Figure 14. Nonlinear (quartic) distribution of uncertainty



(a) Buffer amount w.r.t. merge location ( $x$  represents the distance from the runway: 0 at the runway and 40 at the entry fix.)



(b) Max buffer amount w.r.t. merge location

Figure 11. A total amount of separation with respect to varying merge locations.

is linearly proportional to the route segment length. The plot of constant uncertainty increment and the corresponding optimal merge location is plotted in Figure 12. The maximum of the two solid lines with the symbols represents the traces of the largest separation amount of all the scheduling points. The x-axis represents the location of the downstream merge point. Once the downstream merge point is located at the predicted optimum point, the separation amount at the upper merge point appear to be smaller than the one at the downstream merge point.

On the other hand, if we put more weights in the uncertainty towards the entry fixes, different optimal merge locations are predicted. An assumption of a linear increment as in Figure 13 moves the optimal merge location slightly towards the entry fix, about 10 nmi away from the runway. If we put more weights in the entry fix boundary area as in Figure 14, at a quartic increase rate for example, the optimal merge location falls in the regions farther from the runway, about 23 nmi away.

From these simple models of the uncertainty quantification and propagation, it is concluded that the optimal location of the merge point is where a large amount of uncertainties are present, and the merge point plays a role in reducing the uncertainties in that region by enforcing the pilot's control efforts to meet the suggested STA at that point.

## Conclusions

MILP-based optimization algorithms were used in our scheduling and route assignment problem, and a number of heuristics were introduced into the original formulation to keep its computational cost realizable in the dense terminal airspace operations. FCFS-heuristics and GA-based heuristics were adopted to reduce the computational burden, and the resultant computational costs and scheduling results were compared with the original MILP solutions. The GA-based MILP planner appears to be very efficient without loss of optimality. To take into account uncertainty propagation in the route structure, a dynamic planner framework is developed. The dynamic planner consists of the modules of the planner and the trajectory simulator. The STAs are updated in a dynamic manner via the interaction of the planner and the simulator throughout the whole simulation. An uncertainty model is implemented in the trajectory model of the simulator and the extra separation buffers are added at the scheduling points in the planner. As a practical application of the proposed schedulers, an investigation of the optimal route structure under uncertainties in the extended terminal airspace was carried out. A constant uncertainty increment was assumed along the route, ensuring the uncertainty amount grows linearly in proportion to the route segment length. After analyzing the airspace topologies with varying merge topologies, a route structure having the least maximum separation at the scheduling points has shown the best scheduling performance. These results were validated with a dynamic planner framework with a more reasonable demand set. Finally, airspace topologies with various uncertainty distribution models were tested: a constant, linear and quartic unit-increment of the uncertainty along the route segment. It is shown that the optimal merge point should be positioned to bound the growth of the uncertainty-related separation buffer such that the maximum total separation at any point along the route is minimized. This fact indicates that there exists a likely optimal merge topology. The optimal merge topology is still tuned for a range of uncertainty propagation models while the exact topology did vary.

## References

- [1] Swenson, H., Barhydt, R., and Landis, M., "Next Generation Air Transportation System (NGATS) Air Traffic Management (ATM)-Airspace Project," *NASA Tech. Report*, Ver. 6.0, Moffett Field, CA, 2006.
- [2] Prete, J., Krozel, J., Mitchell, J. S. B., Kim, J., and Zou, J., "Flexible, Performance-based Route Planning for Super-Dense Operations," *AIAA Guidance, Navigation, and Control Conference*, Honolulu, HI, Aug. 2008.
- [3] Isaacson, D. R., Robinson III, J. E., Swenson, H., and Denery, D. G., "A Concept for Robust, High Density Terminal Air Traffic Operations," *10th AIAA Aviation Technology, Integration, And Operations (ATIO) Conference*, Fort Worth, Texas, Sep. 2010.
- [4] Clarke, J. P., Ren, L., Schleicher, D., Thomson, T., Cross, C., Lewis, T. B., "Characterization of and Concepts for Metroplex Operations," *NASA/CR-2010-000000*, 2010.
- [5] Rathinam, S., Montoya, J., and Jung, Y., "An Optimization Model For Reducing Aircraft Taxi Times at the Dallas Fort Worth International Airport," *26th International Congress of the Aeronautical Sciences*, 2008.
- [6] Atkins, S., Capozzi, B., and Choi, S., "Towards Optimal Routing and Scheduling of Metroplex Operations," *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, Sep. 2009, Hilton Head, South Carolina.
- [7] Garcia, J. Berlanga, A., Molina, J. M., Casar, J. R., "Optimization of airport ground operations integrating genetic and dynamic flow management algorithms," *AI Communications*, Vol. 18, N. 2, pp. 143–164, 2005.
- [8] Atkins, S., and Capozzi, B., "Hybrid Optimization Approach To Traffic Management," *10th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, Sep. 2010, Fort Worth, Texas.
- [9] Kaufman, D. E., Nonis, J., and Smith, R. L., "A Mixed Integer Linear Programming Model for Dynamic Route Guidance," *Transportation Research: Part B, Methodology*, Vol. 21 pp. 431–440, 1998.
- [10] Goldberg, D-E., "Genetic Algorithms in Search, Optimization and Machine Learning," *Addison-Wesley*, 1989.
- [11] Barmore, B., Abbott, T., and Krishnamurthy K., "Airborne-Managed Spacing in Multiple Arrival Streams," *24th International Congress of the Aeronautical Sciences*, Sep. 2004, Yokohama, Japan.
- [12] Meyn, Larry. A., and Erzberger, H., "Airport Arrival Capacity Benefits Due to Improved Scheduling Accuracy," *AIAA Aviation, Technology, Integration and Operations Conference (ATIO)*, Arlington, Virginia, September 2005, AIAA 2005-7376.

- [13] Callantine, T. J., "TRAC Trial Planning and Scenario Generation to Support Super-Density Operations Studies," *AIAA Modeling and Simulation Technologies Conference*, Chicago, IL, August 2009, AIAA 2009-5836.